The “Why and How” of a Patent Design-Around Strategy

Table of contents +

A patent conflict doesn’t automatically mean abandoning the product, delaying launch, or accepting unfavorable licensing terms. In suitable circumstances, the team may be able to redesign the relevant product feature so that it falls outside the scope of the patent claims while retaining the functionality of customers’ value.

This process is generally described as “designing around” a patent. It is not a matter of making cosmetic changes, renaming components, or adding secondary features to an otherwise unchanged product. A defensible patent design-around strategy begins with the language of the claims and converts that legal analysis into specific engineering decisions.

Under the all-elements rule applied in US patent infringement analysis, an accused product must contain every claimed element, or an equivalent of every claimed element, for infringement to be established. The US Court of Appeals for the Federal Circuit recently restated on this element-by-element requirement in *Roland Corporation v. in Music Brands, Inc.*

Design-Around as Claim-Level Engineering

Infringement is all about claims and their interpretation. An engineering team cannot determine design-around risk merely by deciding that its product “looks different” from the patented embodiment.

The objective is to identify the technical relationships, structures, or process steps required by a potentially relevant claim. The team must then determine whether one of those requirements can be removed or materially changed without destroying the product’s commercial value. The earlier design-around analysis describes this as spotting the missing element, studying available prior art through an Intellectual property search, and examining amendments that may have narrowed the claims.

A commercially successful redesign must satisfy two tests. It must create sufficient distance from the relevant patent claims, and it must still meet the performance, cost, regulatory, and user requirements of the planned product. This is why design-around projects work best when patent counsel, patent analysts, engineers, and product managers collaborate from the beginning.

Design-Around Workflow

1. Define Design Boundaries

Define which product outcomes are commercially non-negotiable: processing speed, battery life, safety, cost, customer-facing features. This separates the result from the current implementation, often creating room for alternatives. Simultaneously, confirm the patent creates a real obstacle. Check jurisdiction, legal status, ownership, and applicable claims, using a reliable patent database. Not every patent found creates an enforceable barrier in your intended markets.

2. Analyze Claims and Families

Build a claim-to-feature matrix mapping each independent claim limitation to your product component. Check prosecution history, office actions, and amendments. They reveal why claims were narrowed and which alternatives the patentee distinguished from prior art. Review the complete patent family for continuations and divisionals that may contain different claims. This step prevents the eventual redesign from overlooking related obstacles. Patent Analytics can also help teams examine patent families, citations, legal status, and related filing activity more efficiently.

3. Identify Design Alternatives

Identify design levers: remove a component, substitute a mechanism, change operation sequence, move functions between hardware/software, redistribute operations across subsystems, alter data flow. Consult prior art such as expired patents, standards, academic papers, for alternative technical approaches that predate the blocking patent and remain commercially viable. Apply TRIZ principles to identify inventive solutions to the technical contradiction your patent constraint creates. TRIZ’s pattern library often reveals non-obvious alternatives your team hasn’t initially considered.

Check out this Webinar about Patent Circumvention Using TRIZ – YouTube

4. Evaluate Design Options

Compare design alternatives using a structured matrix: claim distance, doctrine-of-equivalents exposure, performance, cost, development time, supply chain, regulatory impact. Test whether your redesign avoids literal claim language and withstands equivalents doctrine. Does the revised mechanism perform substantially the same function in substantially the same way?

5. Verify before design freeze

Map the final production architecture against all relevant claims. Retain claim charts, technical reasoning, test results, and decisions for future teams using suitable intellectual property management software. Features removed during initial redesign may reappear during optimization or supplier substitution.

How Patent Intelligence Helps

Patent intelligence software helps teams find, organize, and monitor the technical and legal information needed for claim mapping, infringement analysis, and legal decisions. Early adoption during product development prevents the costly scenario of discovering blocking patents after design commitments have already been made.

During the initial stages of innovation, semantic and paragraph-similarity searching within advanced patent software can identify patents that describe comparable functions using different terminology. Boolean and classification searching can then narrow the results by component, mechanism, jurisdiction, or technical field.

Within PatSeer, these search approaches can be combined with patent-family and legal-status analysis, AI-generated technical summaries, citation review, patent analytics and shared project workspaces. Teams can retain shortlisted references, organize potentially blocking claims, and make the research available to legal and engineering stakeholders without treating the platform itself as the decision-maker.

Conclusion

A patent design-around strategy is most effective when it is incorporated into product planning rather than initiated only after a cease-and-desist letter or licensing demand. Early analysis gives engineers more architectural freedom and reduces the cost of replacing features that have already been validated, sourced, or integrated.

For R&D and IP teams, the practical question extends beyond identifying blocking patents. The more useful question is whether the commercially important result can be achieved through a technically meaningful route that operates independently of the relevant claims.